Web scraping – Condo Prices in Montreal

php SCHOOL\\_\_\_\_\_FALL2015\PHP\ASSIGNMENT1\scraping.php

**Introduction**

Web scraping is a software technique of extracting information from websites using a program that requests pages like a browser would. The program may simulate user form entry in order to obtain access to the information. The program downloads the pages containing the desired data, parses the HTML, extracts the information and saves/uses it in useful ways. Web scraping is very closely related to web indexing, where a spider extracts information from web sites for the purpose of indexing for a search engine.

There are many different ways to scrape, but we will use a simple one: we will programmatically retrieve web pages by submitting form data, and parse the html using simple DOM parsing.

**Project description**

[Royal Lepage](http://www.royallepage.ca/) publishes their real estate listings in a searchable web site. You will find over 1000 active house listings for Montreal, QC.

There are [legal issues](http://en.wikipedia.org/wiki/Web_scraping#Legal_issues) associated with using data that has been scraped of the web, even if that same information is available by manual browsing. The website has a legal terms of use section as well as a privacy section: neither mention robots nor scraping. The [robots.txt](http://www.royallepage.ca/robots.txt) file disallows the media and profile folders, and asks that robots add a delay of 0.5 seconds between requests (Crawl-delay).

Once you have the legal and ethical go-ahead, you always need to examine the following before starting any web scraping project:

* How are the requests for data made? What kind of http request is required to retrieve data? Is the search criteria in the URL as a querystring (i.e, GET request) or in POST content? Or is Javascript making a request that the browser address bar doesn’t see?
* How are the results returned? Formatted in the HTML page or as JSON?

**Overview**

The focus of this application is to scrape web data to build a database, and once the database is populated, provide a simple web application to perform queries.

You will write a first PHP program (command line/console NOT a web application) that:

* Visits all the pages necessary to extract all the condos for sale in Montreal, QC. Extract information related to the condos (i.e., price, neighborhood, bedrooms, bathrooms) when available
* populates a MySQL database with the results for those condos with all fields

This program will run once and populate the database.

You will then write a second PHP program (web application) that provides query capabilities to find the average price of condos given neighborhood, bedroom and/or bathroom characteristics.

**Project steps**

**Step 1 – Examine how you will request all the pages**

Visit the Royal Lepage website and search for condos for sale in Montreal. You will notice that the URL looks somewhat like this (i.e., the querystring implies the form is submitted using GET)

[http://www.royallepage.ca/search/homes/qc/montreal/?min\_price=0&max\_price=5000000%2B&property\_type=8&lat=45.5016889&lng=-73.56725599999999&display\_type=gallery-view&tier2=False&tier2\_proximity=0&search\_str=Montreal%2C+QC%2C+Canada&beds=0&baths=0&sfproperty\_type[8]=8&transactionType=SALE&address=Montreal&method=homes&address\_type=city&city\_name=Montreal&prov\_code=QC&sortby=low\_to\_high\_price](http://www.royallepage.ca/search/homes/qc/montreal/?min_price=0&max_price=5000000%2B&property_type=8&lat=45.5016889&lng=-73.56725599999999&display_type=gallery-view&tier2=False&tier2_proximity=0&search_str=Montreal%2C+QC%2C+Canada&beds=0&baths=0&sfproperty_type%5b8%5d=8&transactionType=SALE&address=Montreal&method=homes&address_type=city&city_name=Montreal&prov_code=QC&sortby=low_to_high_price)

Note: I played around with the URL and removed unnecessary name-value pairs to get a shorter URL. I also sorted results by price. Use my URL as your starting point.

At the time of writing, there are 45 pages of results. Look at the URL format for subsequent pages. Examine the URL carefully: make sure you understand where to change the URL string in order to traverse through all pages.

**Step 2 – Analyse the html source file**

Look at how the required data (price, neighborhood, bedrooms, bathrooms) is formatted. See if you can see a pattern in the html that will make it easy for you to isolate the condo listings from the rest of the page.

I also want you to be able to isolate the number of pages that you need to iterate through programmatically (i.e., don’t hardcode 45 pages).

**Step 3 – Create the MySQL database and tables**

Create a MySQL database, and the necessary table(s) to store the condo information. Either use MySQL command line tool, or phpMyAdmin (available in the WAMPServer menu), or a PHP script to create the database and tables.

Recall: By default, a MySQL database that you create using WAMPServer’s phpMyAdmin tool is saved on the C: drive. You may want to save your database files on your H: drive or USB key. See Lab 5 for details.Alternatively, you can export / import your database and tables using phpMyAdmin (found in the WAMPServer menu) to have a backup.

NOTE: save the file with the commands used to create your database and table(s). You will submit this!

**Step 4 – PHP Console application to populate the database**

Write a PHP console application. (nb: A php console application looks like a web application, except that it typically does not contain HTML. It is invoked on the command line as php filename.php ) Note: the application can require other files to perform its task.

1. The PHP console application will use the cURL library to make relevant requests to the website. **Be sure** to sleep(1) between requests to add a 1 second crawl delay.
2. On the first page, determine how many more pages there are to scrape. Parse through the html with a DOMDocument and/or DOMXPath.
3. On the first page and all subsequent pages, parse to get all the results (i.e. 20 condo listings/page). You may have to use regular expressions or even good old string parsing to extract the bedroom and bathroom information.
4. When a complete record is found (i.e., all fields have been found), store the parsed data in the database. Use prepared statements!

The console application performs 3 very distinct operations. Use best practice coding standards and MODULARIZE!

**Step 5 – PHP Web application to query the database**

Write a PHP web application that provides basic query capabilities. Feel free to add more, but minimally allow the user to query for:

- the average price of condos based on selection of the neighborhood, bedroom and/or bathroom characteristics (not all fields need to be specified).

- the count of condos available based on selection of the price range, neighborhood, bedroom and/or bathroom characteristics (not all fields need to be specified).

Code this as a self-processing sticky form. Add this acknowledgement: All data © Royal Lepage, 2015

Over-achiever feature: build up your JavaScript toolkit and learn a new feature that you can show off in your CV – add a chart comparing average condo price per neighborhood. Check out [Google Charts](https://developers.google.com/chart/) or [Highcharts](http://www.highcharts.com/) (free for academic use).

**Submittal:**

You must submit soft and hardcopies of:

* the SQL or PHP file used to create your database and table
* all your php scripts
* any other files required (e.g., css, js, or images)